

WHAT IS CLAIMED IS:

1. In a Database Management System (DBMS) holding a base table of information in a Database and using indexes which allow OLE DB applications of a client user to retrieve a selected column of said table without need to
5 access said base table, a system for accessing data from said Database, a method for optimizing a query to said Database comprising the steps of:

10 (a) utilizing a "FIND KEY OF" feature to access said selected column thus to avoid the unnecessary inclusion of columns from a disk file data set;

(b) returning a selected row of information, from said selected column, to said OLE DB application.

2. The method of claim 1 wherein step (a) includes the step of:

(a1) accessing a row of selected data using said "FIND KEY OF" which gathers the column data stored in said index.

5

3. In a Data Management System (DBMS) holding a base table of information in a Database and provides an index structure having index files wherein each index file contains a Key and Pointer which points to a record in said base table, a method for managing database queries by managing overhead operations comprising the steps of:

(a) fetching a limited set of columns from said base table;

(b) issuing a database fetch request from said index structure.

10

4. The method of claim 3 which includes the step of:
(c) enabling OLE DB applications to access selected columns present in said index structure without need to access said base table.

5. The method of claim 3 wherein step (b) includes the step of:

(b1) retrieving column information from said index structure.

6. A system for optimally accessing a database from a Windows based OLE DB application without the need to fetch an entire row from a base table, comprising:

5 (a) client query means for fetching a limited set of columns from a table;

(b) checking means to determine which columns are present in the index structure which spans said table;

10 (c) database fetch request means for fetching a column from only said index structure.

7. The system of claim 6 wherein said client query means (a) includes:

(a1) client query means for fetching a limited set of columns from a table;

5 (a2) means to determine if said Rowset was opened with an Index, if yes then;

(a3) means to determine if the command "FIND KEY OF" is applicable for this Index, and if yes;

10 (a4) means to set the CRowset member variable (m_GetData_CanDefer) to TRUE.

8. The system of claim 6 wherein checking means (b) includes:

5 (b1) creating an accessor means for setting up bindings for each column in a row that needs to be accessed including where the column data is to be laid out in the consumers data buffer and which includes a binding for each column in a row;

10 (b2) means to determine if the Rowset was opened with an Index, and if yes;

(b3) means to check whether said accessor means is a NULL accessor which has no bindings, and if it is a NULL, then;

15 (b4) means to indicate that said Row can be fetched from an Index.

9. The system of claim 8 where means to check (b3) indicates it is NOT a NULL accessor, then includes:

(b3a) means to check all bindings including:

5 (b3a1) means to determine for each binding, if the column is a KEY column or KEYDATA column, and if NOT KEY or KEYDATA column then;

10 (b3a2) means to stop the process as Rows cannot be fetched from Index.

10. The system of claim 9 wherein means (b3a1) is a KEY or KEYDATA column, then includes:

5 (b3a1) means to set CRowset member variable (to indicate deferred column fetch) to enable getting data from the Index.

11. A method for optimizing a query from an OLE DB application in a Data Management System database organized with a base table having rows and columns of data plus an Index Structure with key entries pointing to records in the
5 base table comprising the steps of:

- (a) deciding whether or not it is necessary to access said base table, and if not necessary;
- (b) utilizing a key and/or keydata of an index in said index structure to access said database
10 table columns;
- (c) creating column bindings for only the column data stored in the index structure;
- (d) supplying the data, in response to said OLE DB query, from said index structure.

12. The method of claim 11 which includes the step
of:

5 (e) counting, by said OLE DB application, the
number of rows in a table while traversing one of
the index structures from beginning to end
without creating any column bindings.

13. In a system utilizing an OLE DB to access a Database storing data in tables having rows and columns, said system enabling access to desired data in a rapid efficient manner and comprising:

- 5 (a) means to access desired data from said database by defining column access after first executing access of a desired row.

14. The system of claim 13 wherein said means (a) to access includes:

(a1) means to subsequently access said desired column after accession of said desired row.

5

15. In a Data Management System (DBMS), accessed by applications utilizing the OLE DB interfaces, a system for optimal retrieval of data from a database means organized in base tables of rows and columns correlated to an Index
5 having keys pointing to data records in said base table comprising:

- (a) client-application (100) platform means for accessing and retrieving data from said database (104);
- 10 (b) application program interface means (24) for communication with a Data Provider means (20);
- (c) said data provider means (20) for passing client requests for data to a database server means (104);
- 15 (d) database server means (104) for handling transactions, inquiries and updates.

16. The system of claim 15 wherein said (c) data provider means (20) includes:

5 (c1) data provider object means (25) for passing client requests for data via a transport object (26) and TCP/IP network (22) to a server end module (20B) means;

(c2) said server end module means (20B) including:

10 (c2a) server library means (28) for managing connections from client work stations;

(c2b) worker module means for accessing said database server means (104).

17. The system of claim 15 which includes:

(e) means for fetching rows including:

(e1) means to determine if deferred column fetch is set for the Rowset, and if yes;

5 (e2) means to check if the session-in-transaction and the isolation level is greater than "Read Uncommitted", and if yes;

(e3) means to stop the process since Rows cannot be fetched from said Index.

18. Th system of claim 17 wherein said means (e2) to check indicates the session-in-transaction and isolation level is NOT greater than Read Uncommitted, then said system includes:

5 (e2a) means to send a Defer-Flag to a Worker module in said data provider means to indicate that Rows are to be fetched from Index only.

19. The system of claim 17 which includes:

(e4) means for updating Rows in said DataBase Server means (104) including:

5 (e4a) means for determining if a Row has been fetched in a Deferred Mode, and if yes;

(e4b) means to use the bookmark of the Row to fetch the complete row with all its columns.